

## **REMARKS**

This After Final Amendment is submitted in response to the final Office Action of October 3, 2006 (hereinafter "the Office Action"). Claims 1, 5, 6, 8, 9, and 22-24 remain pending.

### ***Amendment***

Applicants respectfully request entry of this After Final Amendment in accordance with 37 C.F.R. § 1.116 to place this application in condition for allowance.

Claims 1 and 22 are amended to positively recite that the XML document is downloaded into a wireless or hand held mobile device, and that the compiled DTD is executed on a processor of the device. This feature is supported by the specification as originally filed, e.g., in the first four paragraphs of the "Detailed Description" portion of the Application.

### ***Rejection under 35 U.S.C. §101***

Applicant notes with appreciation the withdrawal of the rejection under 35 U.S.C. §101 made in the Office Action of May 19, 2005.

### ***Rejections based on prior art***

Claims 1 and 22 stand rejected under 35 U.S.C. § 102(e) for being anticipated by published U.S. patent application 10/452,282 (Publication No. 2004/0002952) filed by Lee et al, hereinafter referred to as "Lee." Claims 5, 6, 8, 9, 23, and 24 stand rejected under 35 U.S.C. § 103(a) for being unpatentable over Lee in view of U.S. Patent Application 09/753,038 (Publication No. 2001/0054172) filed by Tuatini, hereinafter referred to as "Tuatini." Applicant respectfully traverses because the prior art references do not show or suggest each and every feature set forth in the claims.

By the present Amendment, claims 1 and 22 now positively recite a step of downloading the compiled DTD into a wireless or handheld mobile device. For the reasons explained below, Lee does not disclose a compiled DTD. Tuatini describes a serialization technique that includes a validation component, but is directed toward a service provider (see, e.g., Tuatini, Figure 5 and corresponding text in paragraph 22). There is no suggestion of

using a compiled DTD in a wireless or hand held mobile device as set forth in the presently amended claims.

The Office Action appears to interpret the term, "Document Type Definition" or "DTD" in an overly broad manner. Applicant respectfully points out that, during examination, the Office is obliged to give claim elements the broadest *reasonable* interpretation to claim elements, and this interpretation must be consistent with the interpretation that those skilled in the art would reach. The "broadest reasonable interpretation" should be the "plain meaning" or "ordinary and customary interpretation" of the term, which may be evidenced by a variety of sources, including: the claims themselves, dictionaries and treatises, the written description, the drawings, and the prosecution history. See MPEP 2111.

In the present Application, the phrase, "Document Type Definition" (hereinafter, "DTD") has a specific meaning in the art. As defined in the claims and in Applicant's own disclosure, a DTD "specif[ies] the valid information and arrangement of information in the complex XML document" (page 1, lines 26-28). Lee states: "the DTD is the definition of language. Thus, the DTD has to define the structure of an XML document" (paragraph 19). The second reference (Tuatini) cited by the Examiner similarly states: "The DTD's of a document provide meta data that is used by a parser when parsing the document. The meta data includes allowed sequence and nesting of tags, attribute values, names of external files that may be referenced, the formats of external data that may be referenced, and entities that may be encountered" (paragraph 11). The Article "Data Models" presented in *Computer Science Handbook, Second Ed.*, by Allen B. Tucker, which is attached hereto, states, "The main purpose of a DTD is much like that of a schema: to constrain and type the information present in the document. However, the DTD does not, in fact, constrain types in the sense of basic types like integer or string. Instead, it only constrains the appearance of subelements and attributes within an element. The DTD is primarily a list of rules for what pattern of subelements appear within an element" (p. 52-16, bottom paragraph).

Given that the Applicant, the references relied upon by the Office, and the *Computer Science Handbook* all generally agree on the meaning of the term, DTD, Applicant respectfully submits that the term should be given weight and meaning by the Office. However, the Office Action's statement on page 6, that "Therefore Lee clearly teaches a compiled DTD" implies an *unreasonably broad* interpretation of "DTD." Applicant generally

agrees with the Examiner's understanding of Lee on page 6 of the Office Action, i.e., that Lee teaches an XML validator that receives a DTD document corresponding to an XML document and that the XML validator then applies the XML document to the DTD to verify the validity of the XML document. However, Applicant does not agree with equating the XML validator of Lee to the term "compiled DTD."

Applicant respectfully submits that it is improper to equate "XML validator" of Lee to the claimed "compiled DTD" because the XML validator is not a DTD, as the term would be understood according to its plain meaning given its *broadest reasonable interpretation*. As mentioned above, the term DTD has a specific meaning that is well understood and agreed upon by those skilled in the art, including the primary references relied upon by the Office Action, the present disclosure, and the Encyclopedic *Computer Science Handbook*, as described above. Specifically, that a DTD contains a list of rules and constrains the format of an XML document. It is clear, however, that Lee's XML validator does not contain a list of rules defining elements within an XML document. While the XML validator does receive and parse the DTD, the DTD does not become a part of the XML validator, since the XML validator is separate and distinct from the data upon which it acts. To call the XML validator a compiled DTD is the equivalent of calling the word processing program upon which this Amendment is being drafted, a "Compiled Amendment."

Claim 1 specifically recites, *inter alia*: "accessing a compiled document type definition (DTD) for the XML document, the compiled document type definition being executable program code; and verifying the XML document using the compiled DTD, the verifying comprising generating one of a verified XML output or an error" (claim 1, lines 3-7). Claim 22 includes similar language, but is drawn to a machine-readable medium. For the reasons explained above, Applicant respectfully asserts that Lee does not disclose a compiled DTD. The DTD described in Lee is not an executable program, and the XML verifier, which is an executable program, is not a DTD. Since Lee fails to disclose each and every reference set forth in claims 1 and 22, Applicant respectfully submits that claim 1 is not anticipated by Lee, and that the outstanding rejection should be withdrawn.

Since none of the references teach or suggest downloading a compiled DTD into a wireless or handheld mobile device as now claimed, Applicant respectfully submits that the claims should be allowed.

Applicants respectfully submit that the present Application is now in condition for allowance. A Notice of Allowance is therefore respectfully requested.

If the Examiner has any questions concerning the present amendment, the Examiner is kindly requested to contact the undersigned at (408) 774-6933. If any other fees are due in connection with filing this amendment, the Commissioner is also authorized to charge Deposit Account No. 50-0805 (Order No. SUNMP365). A duplicate copy of the transmittal is enclosed for this purpose.

Respectfully submitted,  
MARTINE PENILLA & GENCARELLA, LLP

A handwritten signature in cursive script, reading "Leonard Heyman", with a long horizontal flourish extending to the right.

Leonard Heyman  
Reg. No. 40, 418

710 Lakeway Drive, Suite 200  
Sunnyvale, CA 94085  
Telephone: (408) 749-6900  
Facsimile: (408) 749-6901  
**Customer Number 32291**